第一大题为概念简单题，选 5 个
第二到第十题为问答和计算题，按之前的卷子应该是六道题目比较合适
第十一题为开放式问答题
所有题目的考察点已经在题目前标出

一、简答题 （概念题选五个）

1. 两个仅包含二元属性的对象之间的相似性度量称为相似系数，简述三种(含)以上相似系数的计算方法与应用场景

   杰卡德相似系数(Jaccard Similarity Coefficient)

   Jaccard(a,b)=$\frac{f11}{f11+f10+f01}$

   只关心个体间的各维度值是否一致这个问题

   皮尔逊相关系数(Pearson Correlation Coefficient)

   $$\rho_{X,Y}=\frac{\operatorname{cov}(X,Y)}{\sigma_X\sigma_Y}=\frac{E((X-\mu_X)(Y-\mu_Y))}{\sigma_X\sigma_Y}=\frac{E(XY)-E(X)E(Y)}{\sqrt{E(X^2)-E^2(X)}\sqrt{E(Y^2)-E^2(Y)}}$$

   数据存在“分数膨胀“问题

   余弦相似度(Cosine Similarity)

   Cos(a,b)=$\frac{ab}{|a||b|}$

   数据是稀疏的，就使用余弦相似度

2. 简述关联规则中支持度和置信度的概念，并解释为什么采用这两种度量来表示关联规则的强度

   Support（支持度）：表示同时包含 A 和 B 的事务占所有事务的比例

   Confidence（可信度）：表示包含 A 的事务中同时包含 B 的事务的比例，即同时包含 A 和 B 的事务占包含 A 的事务的比例。

3. 简述 Apriori 算法的计算复杂度受哪些因素影响，并加以解释

   项数（维度）
   事务平均宽度
   事务数
   支持度阀值

   ## 4.2. 时间复杂度

   频繁 1-项集的产生 对于每个事务，需要更新事务中出现的每个项的支持度计数。假定 w 为事务的平均宽度，则该操作需要的时间为 O(Nw)，其中 N 为事务的总数。

候选的产生 为了产生候选 k-项集，需要合并一对频繁(k-1)-项集，确定它们是否至少有 k-2 个项相同。每次合并操作最多需要 k-2 次相等比较。在最好情况下，每次合并都产生一个可行的候选 k-项集；在最坏的情况下，算法必须合并上次迭代发现的每对频繁(k-1)-项集。因此，合并频繁项集的总开销为：

$$\sum_{k=2}^{w}(k-2)|C_k| < \text{合并开销} < \sum_{k=2}^{w}(k-2)|F_{k-1}|^2$$

Hash 树在候选产生时构造，以存放候选项集。由于 Hash 树的最大深度为 k，将候选项集散列到 Hash 树的开销为$O(\sum_{k=2}^{w} k|C_k|)$。在候选项剪枝的过程中，需要检验每个候选 k-项集的 k-2 个子集是否频繁。由于在 Hash 树上查找一个候选的花费是 O(k)，因此候选剪枝需要的时间是$O(\sum_{k=2}^{w} k(k-2)|C_k|)$。

支持度计数 每个长度为|t|的事务将产生$C_{|t|}^{k}$个 k-项集。这也是每个事务遍历 Hash 树的有效次数。支持度计数的开销为$O(N\sum_k C_w^k \alpha_k)$，其中 w 是事务的最大宽度，$\alpha_k$是更新 Hash 树中一个候选 k-项集的支持度计数的开销。

4. 在分类算法的评价指标中，recall 和 precision 分别是什么含义
   查全率，查准率
5. 请写出构建决策树时不纯度度量的三种指标
   Ent(d)信息熵，信息增益，增益率，基尼系数
6. SVM 中核函数的作用是什么？
   非线性可分时，增加维度，减少计算代价
7. 介绍 k-means 算法对初始点敏感的缺点（可以图示辅助分析）
   朴素的初始化方法是直接在数据点中随机抽取 k 个作为聚类初始中心点。不够好的初始值可能造成收敛速度很慢或者聚类失败。

8. 传统的推荐系统算法主要是哪两种？

   1. 基于用户行为数据的协同过滤算法

   2. 基于内容数据的过滤算法

9.请写出两个 social network 方向的研究内容，如影响力分析
   静态网络的随机产生机制
   研究一个静态网络和 node 变量的互动关系

二、（关联规则）Apriori 算法使用产生—计数的策略找出频繁项集。通过合并一对大小为 k 的频繁项集得到一个大小为 k+1 的候选项集(称作候选产生步骤)。在候选项集剪枝步骤中，如果一个候选项集的任何一个子集是不频繁的，则该候选项集将被丢弃。假定将 Apriori 算法用于表中所示数据集，最小支持度为 30%，即任何一个项集在少于 3 个事务中出现就被认为是非频繁的。

事务 ID	购买项
1	{a,b,d,e}
2	{b,c,d}
3	{a,b,d,e}
4	{a,c,d,e}

5	{b,c,d,e}
6	{b,d,e}
7	{c,d}
8	{a,b,c}
9	{a,d,e}
10	{b,d}

(a) 画出表示表中所示数据的项集格，用下面的字母标记格中的每个结点。

- **N**：如果该项集被 Apriori 算法认为不是候选项集。一个项集不是候选项集有两种可能的原因：它没有在候选项集产生步骤产生，或它在候选项集产生步骤产生，但是由于它的一个子集是非频繁的而在候选项集简直步骤被丢掉
- **F**：如果该候选项集被 Apriori 算法认为是非频繁的
- **I**：如果经过支持度计数后，该候选项集被发现是非频繁的

(b) 频繁项集的百分比是多少？（考虑格中所有的项集）

(c) 对于该数据集，Apriori 算法的剪枝率是多少？（剪枝率定义为由于如下原因不认为是候选的项集所占的百分比：在候选项集产生时未被产生，或在候选剪枝步骤被丢掉）

(d) 假警告率是多少？（假警告率是指经过支持度计算后被发现是非频繁的候选项集所占的百分比）

三、（朴素贝叶斯）试由下表的训练数据学习一个朴素贝叶斯分类器并确定 x = （2，S）T 的类判别结果 y。表中 $X^{(1)}$，$X^{(2)}$ 为特征，Y 为类标记。

	1	2	3	4	5	6	7	8
$X^{(1)}$	1	1	1	2	2	2	3	3
$X^{(2)}$	S	M	M	S	M	M	L	M
Y	-1	-1	1	1	-1	1	1	1

四、（SVM）已知正例点 $x_1 = (2.5, 2.5)^T$，$x_2 = (5, 2)^T$，负例点 $x_3 = (1.5, 1.5)^T$，试用 SVM 对其进行分类，求最大间隔分离超平面，并指出所有的支持向量。

五、（决策树）下表是一个由 15 个贷款申请训练数据，数据包括贷款申请人的四个特征属性：分别是年龄，是否有工作，是否有自己的房子以及信贷情况，表的最后一列为类别，是否同意贷款。

ID	年龄	有工作	有自己的房子	信贷情况	类别
1	青年	否	否	一般	否
2	青年	否	否	好	否
3	青年	是	否	好	是
4	青年	是	是	一般	是
5	青年	否	否	一般	否
6	中年	否	否	一般	否
7	中年	否	否	好	否
8	中年	是	是	好	是
9	中年	否	是	非常好	是

10	中年	否	是	非常好	是
11	老年	否	是	非常好	是
12	老年	否	是	好	是
13	老年	是	否	好	是
14	老年	是	否	非常好	是
15	老年	否	否	一般	否

1）请根据上表的训练数据，以错误率作为划分标准来构建预测是否进行放贷的决策树。

2）按照所构建的决策树，对属性值为（中年，无工作，无自己的房子，信贷情况好）的申请者是否进行放贷

3） 在构建决策树的时候，可能会出现过拟合的问题，有哪些方法可以避免或者解决？

4） 对于含有连续型属性的样本数据，决策树有哪些处理方法？

7. 下表汇总了具有三个属性 *A*, *B*, *C*，以及两个类标号＋，－的数据集。建立一棵两层决策树。

A	*B*	*C*	实例数	
			+	−
T	T	T	5	0
F	T	T	0	20
T	F	T	20	0
F	F	T	0	5
T	T	F	0	0
F	T	F	25	0
T	F	F	0	0
F	F	F	0	25

(a) 根据分类错误率，哪个属性应当选作第一个划分属性？对每个属性，给出相依表和分类错误率的增益。

(b) 对根结点的两个子女重复以上问题。

(c) 最终的决策树错误分类的实例数是多少？

(d) 使用 *C* 作为划分属性，重复(a)、(b)和(c)。

(e) 使用(c)和(d)中的结果分析决策树归纳算法贪心的本质。

Answer:

The error rate for the data without partitioning on any attribute is

$$E_{orig} = 1 - \max(\frac{50}{100}, \frac{50}{100}) = \frac{50}{100}.$$

After splitting on attribute A, the gain in error rate is:

	$A = T$	$A = F$
+	25	25
−	0	50

$$E_{A=T} = 1 - \max(\frac{25}{25}, \frac{0}{25}) = \frac{0}{25} = 0$$
$$E_{A=F} = 1 - \max(\frac{25}{75}, \frac{50}{75}) = \frac{25}{75}$$
$$\Delta_A = E_{orig} - \frac{25}{100}E_{A=T} - \frac{75}{100}E_{A=F} = \frac{25}{100}$$

After splitting on attribute B, the gain in error rate is:

	$B = T$	$B = F$
+	30	20
−	20	30

$$E_{B=T} = \frac{20}{50}$$
$$E_{B=F} = \frac{20}{50}$$
$$\Delta_B = E_{orig} - \frac{50}{100}E_{B=T} - \frac{50}{100}E_{B=F} = \frac{10}{100}$$

After splitting on attribute C, the gain in error rate is:

	$C = T$	$C = F$
+	25	25
−	25	25

$$E_{C=T} = \frac{25}{50}$$
$$E_{C=F} = \frac{25}{50}$$
$$\Delta_C = E_{orig} - \frac{50}{100}E_{C=T} - \frac{50}{100}E_{C=F} = \frac{0}{100} = 0$$

The algorithm chooses attribute A because it has the highest gain.

(b) Repeat for the two children of the root node.

Answer:

Because the $A = T$ child node is pure, no further splitting is needed.
For the $A = F$ child node, the distribution of training instances is:

B	C	Class label	
		+	−
T	T	0	20
F	T	0	5
T	F	25	0
F	F	0	25

The classification error of the $A = F$ child node is:

$$E_{orig} = \frac{25}{75}$$

After splitting on attribute B, the gain in error rate is:

	$B=T$	$B=F$
+	25	0
−	20	30

$$E_{B=T} = \frac{20}{45}$$
$$E_{B=F} = 0$$
$$\Delta_B = E_{orig} - \frac{45}{75}E_{B=T} - \frac{20}{75}E_{B=F} = \frac{5}{75}$$

After splitting on attribute C, the gain in error rate is:

	$C=T$	$C=F$
+	0	25
−	25	25

$$E_{C=T} = \frac{0}{25}$$
$$E_{C=F} = \frac{25}{50}$$
$$\Delta_C = E_{orig} - \frac{25}{75}E_{C=T} - \frac{50}{75}E_{C=F} = 0$$

The split will be made on attribute B.

(c) How many instances are misclassified by the resulting decision tree?

Answer:

20 instances are misclassified. (The error rate is $\frac{20}{100}$.)

(d) Repeat parts (a), (b), and (c) using C as the splitting attribute.

Answer:

For the $C = T$ child node, the error rate before splitting is:
$E_{orig} = \frac{25}{50}$.
After splitting on attribute A, the gain in error rate is:

	$A=T$	$A=F$
+	25	0
−	0	25

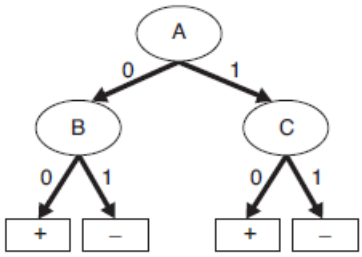
$$E_{A=T} = 0$$
$$E_{A=F} = 0$$
$$\Delta_A = \frac{25}{50}$$

After splitting on attribute B, the gain in error rate is:

	$B=T$	$B=F$
+	5	20
−	20	5

$$E_{B=T} = \frac{5}{25}$$
$$E_{B=F} = \frac{5}{25}$$
$$\Delta_B = \frac{15}{50}$$

Therefore, A is chosen as the splitting attribute.



Training:

Instance	A	B	C	Class
1	0	0	0	+
2	0	0	1	+
3	0	1	0	+
4	0	1	1	–
5	1	0	0	+
6	1	0	0	+
7	1	1	0	–
8	1	0	1	+
9	1	1	0	–
10	1	1	0	–

Validation:

Instance	A	B	C	Class
11	0	0	0	+
12	0	1	1	+
13	1	1	0	+
14	1	0	1	–
15	1	0	0	+

Figure 4.2. Decision tree and data sets for Exercise 8.

For the $C = F$ child, the error rate before splitting is: $E_{orig} = \frac{25}{50}$.
After splitting on attribute A, the error rate is:

	$A = T$	$A = F$
+	0	25
–	0	25

$$E_{A=T} = 0$$
$$E_{A=F} = \frac{25}{50}$$
$$\Delta_A = 0$$

After splitting on attribute B, the error rate is:

	$B = T$	$B = F$
+	25	0
–	0	25

$$E_{B=T} = 0$$
$$E_{B=F} = 0$$
$$\Delta_B = \frac{25}{50}$$

Therefore, B is used as the splitting attribute.
The overall error rate of the induced tree is 0.

(e) Use the results in parts (c) and (d) to conclude about the greedy nature of the decision tree induction algorithm.
The greedy heuristic does not necessarily lead to the best tree.

六、聚类分析（聚类）

(1) 在聚类分析中，传统的 K-means 算法都有哪些局限性？有哪些相应的改进方法？

（1）对于离群点和孤立点敏感；

(2)k 值选择;

(3)初始聚类中心的选择；

(4)只能发现球状簇。

首先针对（1），对于离群点和孤立点敏感，如何解决？笔者在前面的一篇博客

中，提到过离群点检测的 LOF 算法，通过去除离群点后再聚类，可以减少离群点

和孤立点对于聚类效果的影响。

针对（2）k 值的选择问题，在安徽大学李芳的硕士论文中提到了 k-Means 算法的 k 值自适应优化方法。下面将针对该方法进行总结。

首先该算法针对 K-means 算法的以下主要缺点进行了改进：

1）必须首先给出 k（要生成的簇的数目），k 值很难选择。事先并不知道给定的数据应该被分成什么类别才是最优的。

2)初始聚类中心的选择是 K-means 的一个问题。

李芳设计的算法思路是这样的：可以通过在一开始给定一个适合的数值给 k，通过一次 K-means 算法得到一次聚类中心。对于得到的聚类中心，根据得到的 k 个聚类的距离情况，合并距离最近的类，因此聚类中心数减小，当将其用于下次聚类时，相应的聚类数目也减小了，最终得到合适数目的聚类数。可以通过一个评判值 E 来确定聚类数得到一个合适的位置停下来，而不继续合并聚类中心。重复上述循环，直至评判函数收敛为止，最终得到较优聚类数的聚类结果。

针对（3）对初始聚类中心的选择的优化。一句话概括为：选择批次距离尽可能远的 K 个点。具体选择步骤如下。首先随机选择一个点作为第一个初始类簇中心点，然后选择距离该点最远的那个点作为第二个初始类簇中心点，然后再选择距离前两个点的最近距离最大的点作为第三个初始类簇的中心点，以此类推，直至选出 K 个初始类簇中心点。

针对（4）只能获取球状簇的根本原因在于，距离度量的方式。在李荟娆的硕士论文 K_means 聚类方法的改进及其应用中提到了基于 2 种测度的改进，改进后，可以去发现非负、类椭圆形的数据。但是对于这一改进，个人认为，并没有很好的解决 K-means 在这一缺点的问题，如果数据集中有不规则的数据，往往通过基于密度的聚类算法更加适合，比如 DESCAN 算法。

(2) 请简要描述聚类与关联分析的主要相似点和不同点。

聚类。聚类类似于分类，但与分类的目的不同，是针对数据的相似性和差异性将一组数据分为几个类别。属于同一类别的数据间的相似性很大，但不同类别之间数据的相似性很小，跨类的数据关联性很低。

关联规则。关联规则是隐藏在数据项之间的关联或相互关系，即可以根据一个数据项的出现推导出其他数据项的出现。关联规则的挖掘过程主要包括两个阶段：第一阶段为从海量原始数据中找出所有的高频项目组;第二极端为从这些高频项目组产生关联规则。关联规则挖掘技术已经被广泛应用于金融行业企业中用以预测客户的需求，各银行在自己的 ATM 机上通过捆绑客户可能感兴趣的信息供用户了解并获取相应信息来改善自身的营销。

(3) 请举出一个采用聚类作为主要的数据挖掘方法的实际应用例子。

图像分割；在商业上，聚类分析被用来发现不同的客户群，并且通过购买模式刻画不同的客户群的特征。聚类分析是细分市场的有效工具，同时也可用于研究消费者行为，寻找新的潜在市场、选择实验的市场，并作为多元分析的预处理。在生物上，聚类分析被用来动植物分类和对基因进行分类，获取对种群固有结构的认识。在地理上，聚类能够帮助在地球中被观察的数据库商趋于的相似性。在保险行业上，聚类分析通过一个高的平均消费来鉴定汽车保险单持有者的分组，同时根据住宅类型，价值，地理位置来鉴定一个城市的房产分组。在因特网应用上，聚类分析被用来在网上进行文档归类来修复信息。在电子商务上，聚类分析在电子商务中网站建设数据挖掘中也是很重要的一个方面，通过分组聚类出具有相似浏览行为的客户，并分析客户的共同特征，可以更好的帮助电子商务的用户了解自己的客户，向客户提供更合适的服务。

七、（决策树）证明：在决策树分类方法中，将结点划分为更小的后继结点后，结点熵不会增加

3.5 证明：将结点划分为更小的后续结点之后，结点熵不会增加。

证明：根据定义可知，熵值越大，类分布越均匀；熵值越小，类分布越不平衡。假设原有的结点属于各个类的概率都相等，熵值为 1，则分出来的后续结点在各个类上均匀分布，此时熵值为 1，即熵值不变。假设原有的结点属于个各类的概率不等，因而分出来的后续结点不均匀地分布在各个类上，则此时的分类比原有的分类更不均匀，故熵值减少。

八、（效果评价 ROC）请评价两个分类器 M1 和 M2 的性能。所选择的测试集包含 26 个二值属性，记作 A 到 Z。

表中是模型应用到测试集时得到的后验概率（图中只显示正类的后验概率）。因为这是二类问题，所以 P(-)=1-P(+),P(-|A,…,Z)=1-P(+|A,…,Z)。假设需要从正类中检测实例

(a) 画出 M1 和 M2 的 ROC 曲线（画在一幅图中）。哪个模型更多？给出理由

(b) 对模型 M1，假设截止阈值 t=0.5.换句话说，任何后验概率大于 t 的测试实例都被看作正例。计算模型在此阈值下的 precision，recall 和 F-score

(c) 对模型 M2 使用相同的截止阈值重复（b）的分析。比较两个模型的 F-score，哪个模型更好？所得结果与从 ROC 曲线中得到的结论一致吗》

(d) 使用阈值 t=0.1 对模型 M2 重复（b）的分析。t=0.5 和 t=0.1 哪一个阈值更好?该结果和你从 ROC 曲线中得到的一致吗？

实例	真实类	P(+|A,…,Z,M1)	P(-|A,…,Z,M2)
1	+	0.73	0.61

2	+	0.69	0.03
3	-	0.44	0.68
4	-	0.55	0.31
5	+	0.67	0.45
6	+	0.47	0.09
7	-	0.08	0.38
8	-	0.15	0.05
9	+	0.45	0.01
10	-	0.35	0.04

九（频繁项）考虑下面的候选 3-项集的集合：{1，2，3}，{1，2，5}，{1，2，6}，{1，3，4}，{2，3，4}，{2，4，5}，{3，4，6}，{4，5，6}

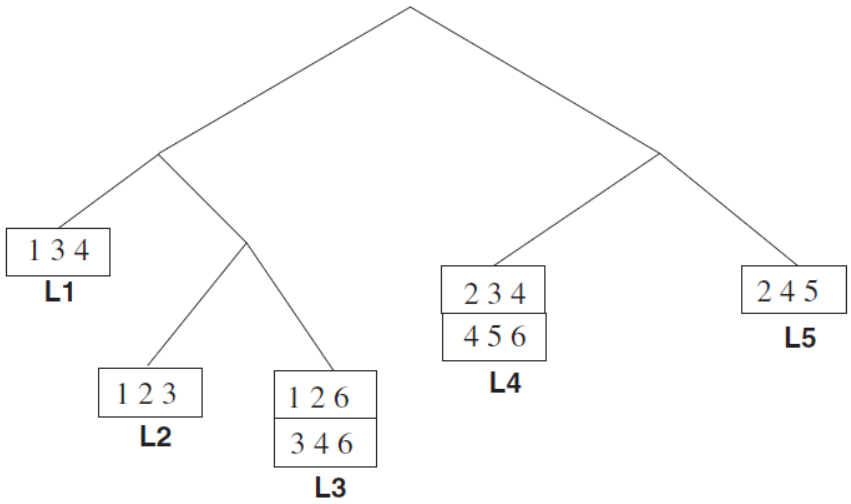
(a) 构造以上候选 3-项集的 Hash 树，假定 Hash 树使用这样一个 Hash 函数：所有奇数项都被散列到节点的左子女，所有的偶数项都被散列到右子女。一个候选 k-项集按照如下方法被插入到 Hash 树中：散列候选项集中的每个相继项，然后再按照散列值到相应的分支。一旦到达叶节点，候选项集将按照下面的条件插入：

- 条件 1：如果该叶节点的深度等于 k（假设根节点的深度为 0），则不管该节点已经存储了多少项集，将该候选插入该节点
- 条件 2：如果该叶节点的深度小于 k，则只要该节点存储的项集数不超过 maxsize，就把它插入到该叶节点。这里，假定 maxsize 为 2
- 条件 3：如果该叶节点的深度小于 k 且该节点已存储的项集数量超过 maxsize，则这个叶节点转变为内部节点，并创建新的叶节点作为老的叶节点的子女。先前老叶节点中存放的候选项集按照散列值分布到其子女中。新的候选项集也按照散列值存储到相应的叶节点

（b）候选 Hash 树中共多少个叶节点，多少个内部节点？

（c）考虑一个包含项集{1，2，3，4，5，6}的事务，使用（a）所创建的 Hash 树，则该事务要检查哪些叶节点？该事务包含哪些候选 3-项集

习题 10



(b) How many leaf nodes are there in the candidate hash tree? How many internal nodes are there?

Answer: There are 5 leaf nodes and 4 internal nodes.

(c) Consider a transaction that contains the following items: $\{1, 2, 3, 5, 6\}$. Using the hash tree constructed in part (a), which leaf nodes will be checked against the transaction? What are the candidate 3-itemsets contained in the transaction?

Answer: The leaf nodes L1, L2, L3, and L4 will be checked against the transaction. The candidate itemsets contained in the transaction include {1,2,3} and {1,2,6}.

十、(ensemble 组合方法)请简述构建组合（集成）分类器的几种方法，并说明集成分类器能够改善分类器性能的原因。

Boosting , Bagging and 随机森林 ，组合多种方式，看书上。

十一、（开放课题）现有一个城市的数据集，包括交通卡、交通事故、出租车轨迹、公交车运行、地铁运行、空气质量、气象检测、新浪微博等（具体特征如下表）。

请利用你所学过的机器学习和数据挖掘的方法解决预测该城市空气质量的问题：

（1） 哪些数据或者特征可能用到，并简要说明原因

（2） 可以使用所学过的哪些机器学习方法解决该问题？

（3） 请简要给出一个解决方案（最大限度地利用现有数据）。

序号	数据集名称	具体数据项
1	城市道路交通指数	状态、区域、当前指数、参考指数、指数差值
2	地铁运行数据	线路、车站、换乘站数据、首末班车各站时刻表数据、站间运行时间数据、限流车站、封站数据、路网票价矩阵、列车实时到发站台时刻、线路拥挤及阻塞数据、出入口、厕所、残疾电梯数据
3	一卡通乘客刷卡数据	卡号、交易日期、交易时间、线路/地铁站点名称、行业名称（公交、地铁、出租、轮渡、P+R 停车场）、交易金额、交易性质（非优惠、优惠、无）
4	浦东公交车实时数据	设备号码,线路编码,站点编码,协议编号,进出站状态,方向,车载上报时间、编码对应表
5	强生出租汽车行车数据	车辆 ID、GPS 时间、经纬度、速度、卫星颗数、营运状态高架状态、制动状态
6	空气质量状况	序号，日期，PM2.5，PM10，O3，SO2，NO2，CO，AQI，质量评价，首要污染物
7	气象数据	日期、时间、监测点、天气类型、温度、风速、风向、降水量
8	道路事故数据	事故 ID、事故类型、事故地点、事故时间
9	高架匝道关闭数据	匝道 ID、位置信息、关闭时间、开放时间
10	新浪微博交通数据	涵盖路况、交通工具、天气等与交通相关的关键词的微博信息